Cloud Networking (VITMMA02)
OpenStack

Markosz Maliosz PhD

Department of Telecommunications and Media Informatics
Faculty of Electrical Engineering and Informatics
Budapest University of Technology and Economics

Spring 2017

OpenStack

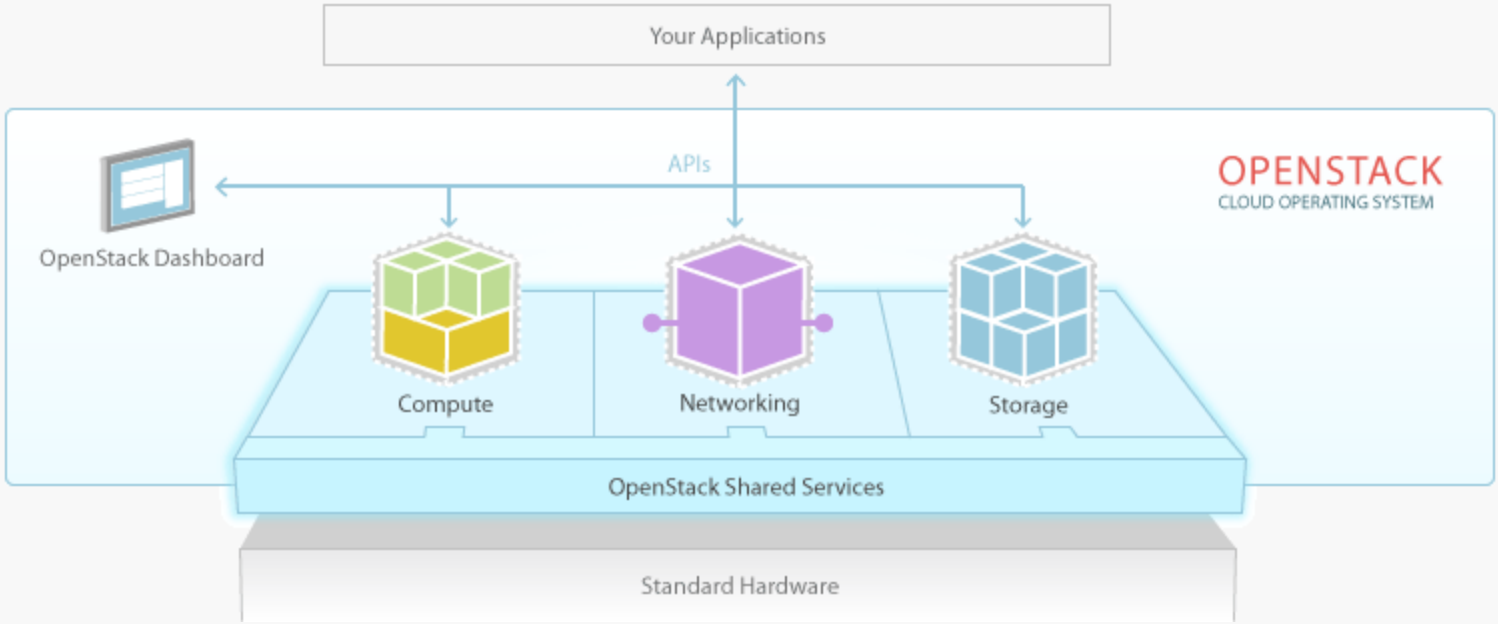
- Free and open source IaaS cloud platform
  - manages processing, storage, network resources
  - on commodity hardware
  - flexible configuration options
- Based on collection of open source software
- started as a joint project of Rackspace and NASA in 2010
  - AT&T, IBM, HP, RedHat, Cisco, Dell, stb. http://www.openstack.org/foundation/companies/
- written in python
- well documented
- modular architecture
- ApacheLicense 2.0

Can	Cannot	Must
Commercial Use	Hold Liable	Include Copyright
Modify	Use Trademark	Include License
Distribute		State Changes
Sublicense		Include Notice
Private Use		
Use Patent Claims		
Place Warranty		

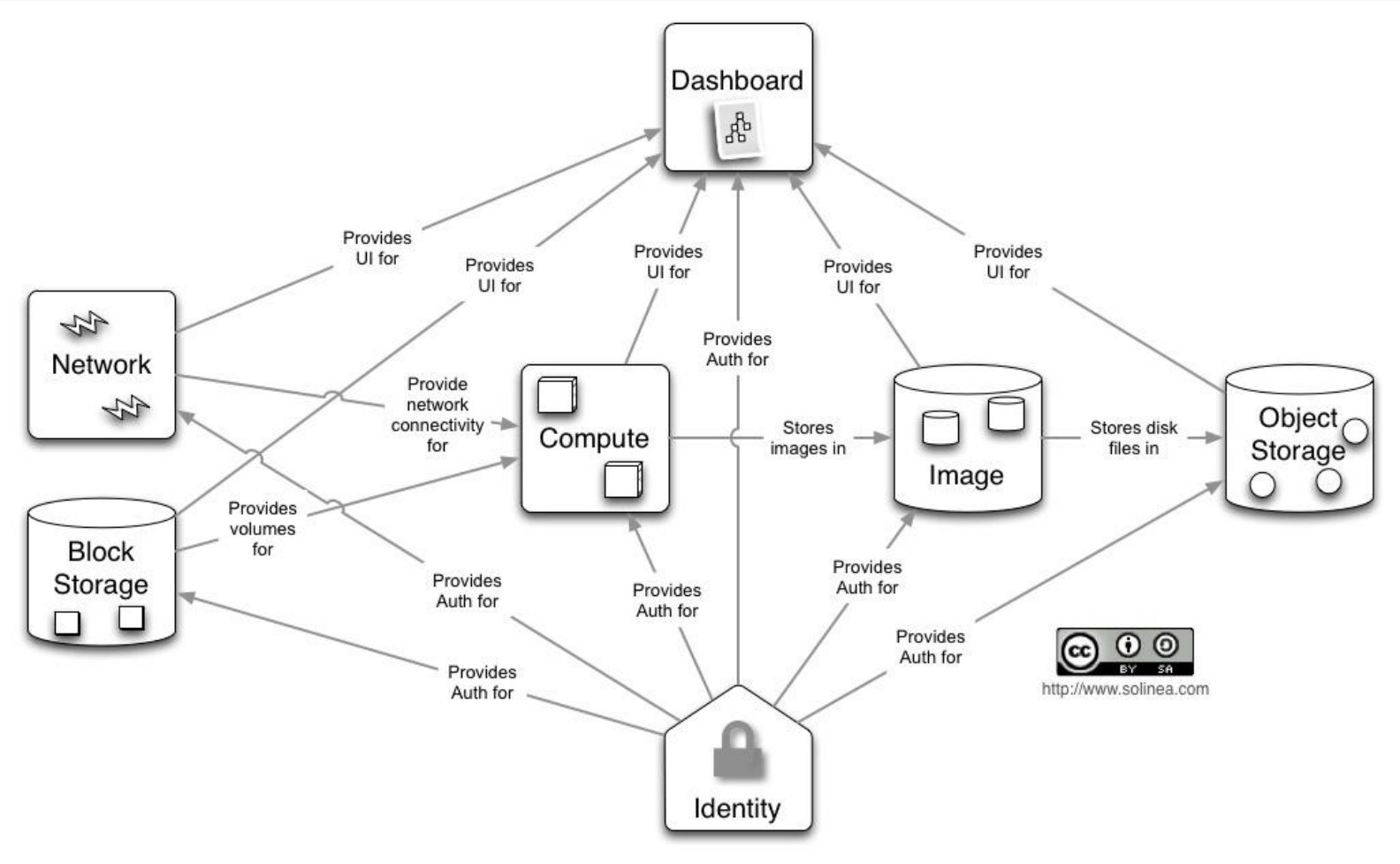
Releases

Release Name	Release Date	Included Components
Austin	21 October 2010	Nova, Swift
Bexar	3 February 2011	Nova, Glance, Swift
Cactus	15 April 2011	Nova, Glance, Swift
Diablo	22 September 2011	Nova, Glance, Swift
Essex	5 April 2012	Nova, Glance, Swift, Horizon, Keystone
Folsom	27 September 2012	Nova, Glance, Swift, Horizon, Keystone, Quantum, Cinder
Grizzly	4 April 2013	Nova, Glance, Swift, Horizon, Keystone, Quantum, Cinder
Havana	17 October 2013	Nova, Glance, Swift, Horizon, Keystone, Neutron, Cinder, Ceilometer, Heat
Icehouse	17 April 2014	Nova, Glance, Swift, Horizon, Keystone, Neutron, Cinder, Ceilometer, Heat, Trove
Juno	October 2014	Nova, Glance, Swift, Horizon, Keystone, Neutron, Cinder, Ceilometer, Heat, Trove (DBaaS), Sahara (data processing)
Kilo	April 2015	Nova, Glance, Swift, Horizon, Keystone, Neutron, Cinder, Ceilometer, Heat, Trove, Sahara, Ironic (bare metal)
Liberty	October 2015	+: Searchlight, Designate (DNS), Zaqar (messaging), Barbican (key manager), Manila (shared file system)
Mitaka	April 2016	+: cloudkitty (billing and charging), freezer (backup and recovery), magnum (container orchestration), monasca (monitoring), senlin (clustering), solum (app. lifecycle framework), tacker (NFV)
Newton	October 2016	+: panko (telemetry), virtage (Root Cause Analysis), watcher (resource optimization)

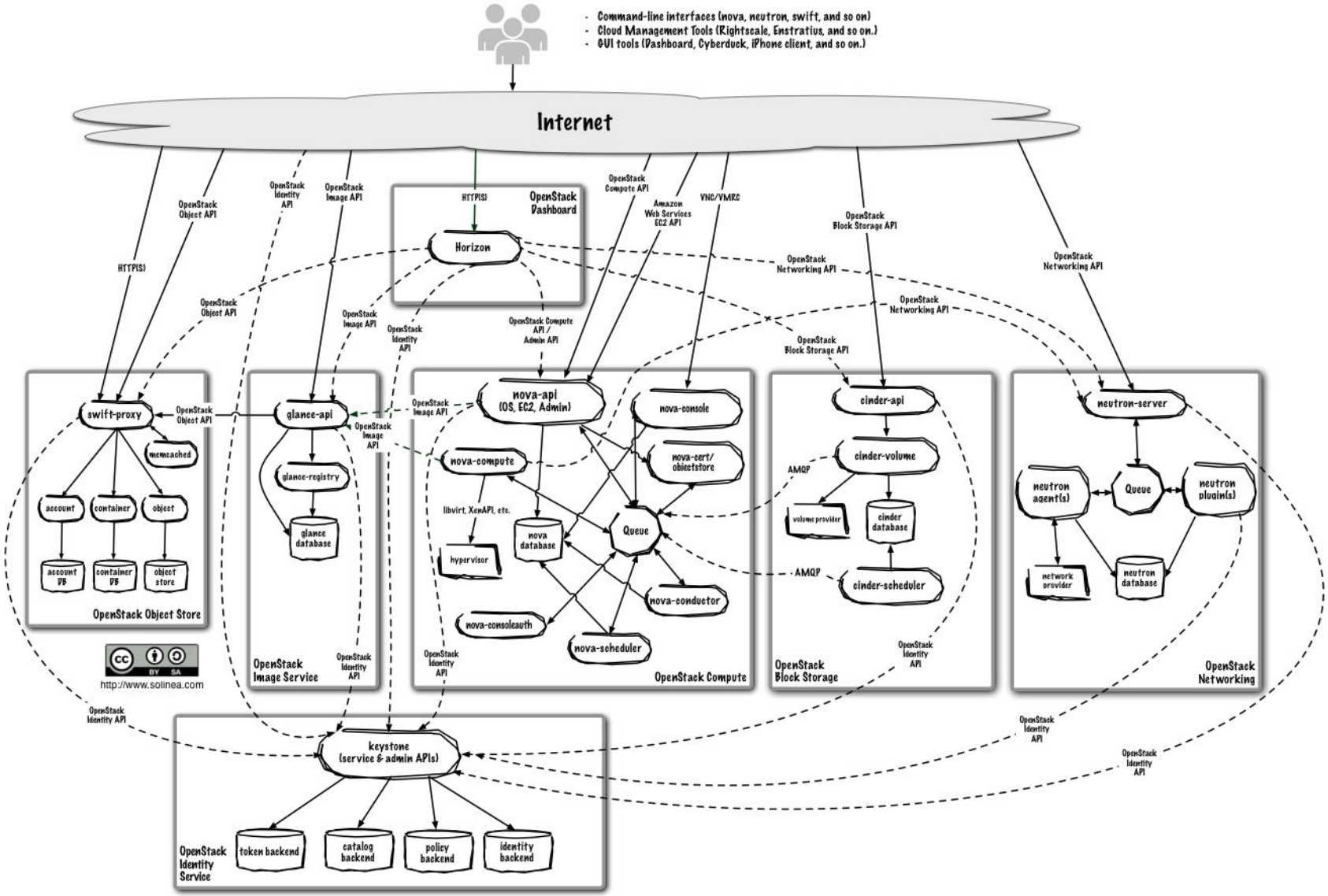
Architectural Overview



Looking into a little bit closer



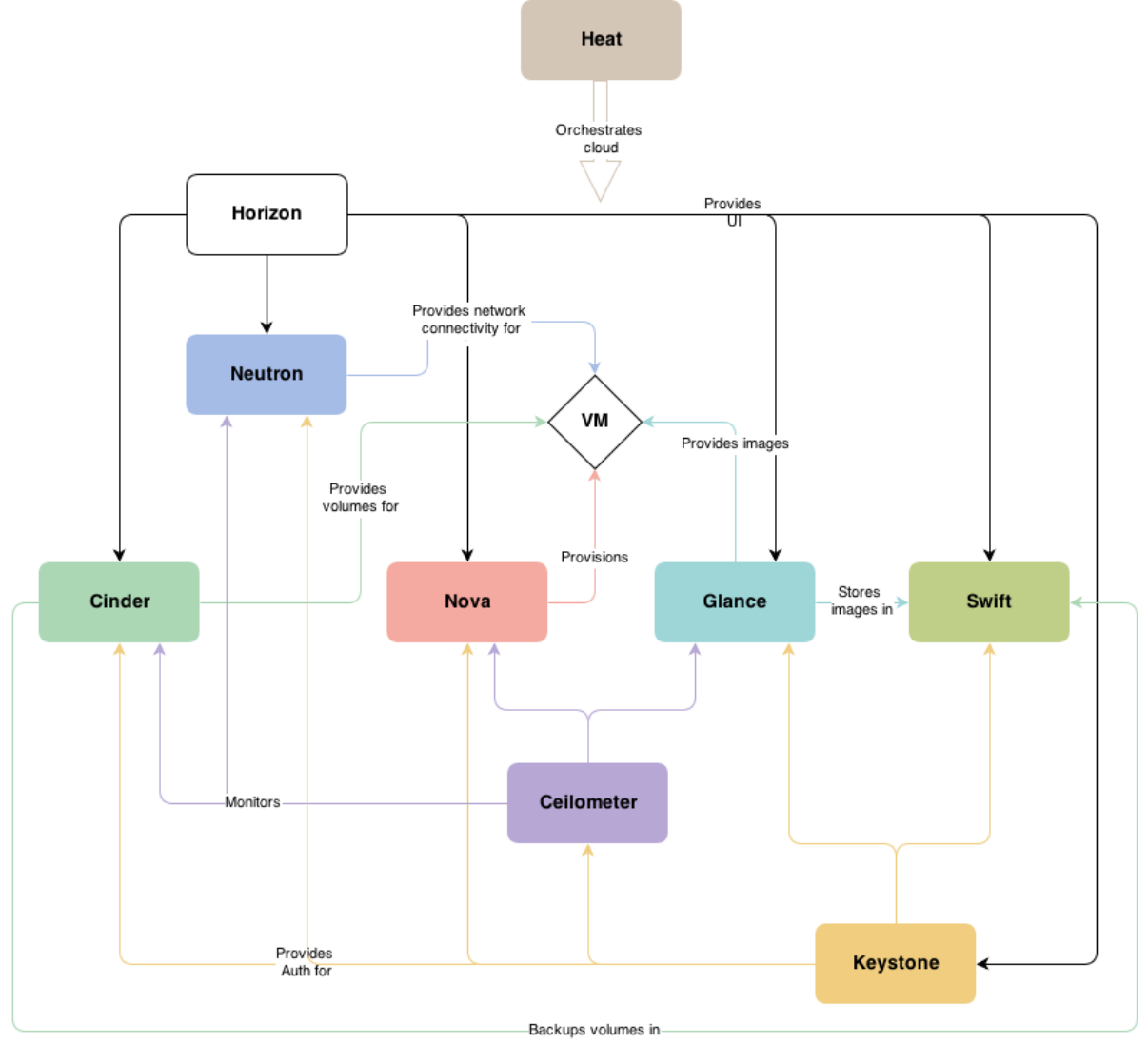
OpenStack is not that simple



OpenStack components

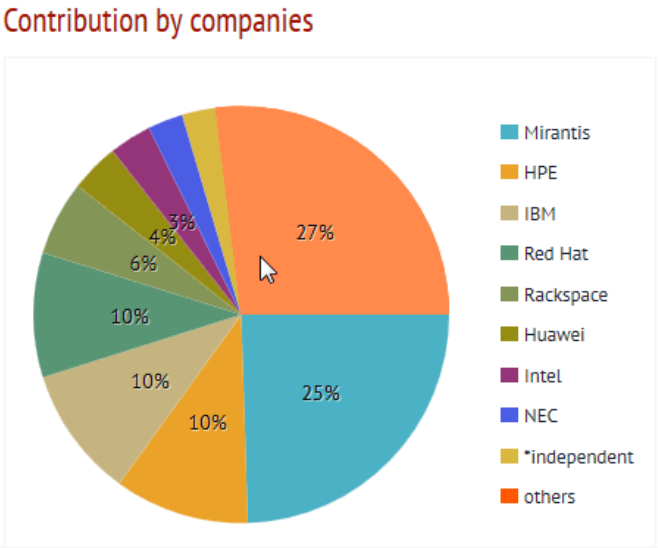
- Dashboard ("Horizon"): web interface
- Compute ("Nova"): running VMs, reading VM images, storing VM images with the help of Image service ("Glance")
- Network ("Neutron"): provides virtual networking for a Compute nodes
- Block Storage ("Cinder"): virtualizes the management of block storage for Compute nodes
- Object Storage ("Swift"): store and retrieve data objects
- Image ("Glance"): VM image management, storage with e.g. Object Storage ("Swift")
- Identity ("Keystone"): central authentication
- Telemetry (Ceilometer): usage monitoring
- Orchestration (Heat): automated VM management
- Database as a Service (Trove)
- etc.

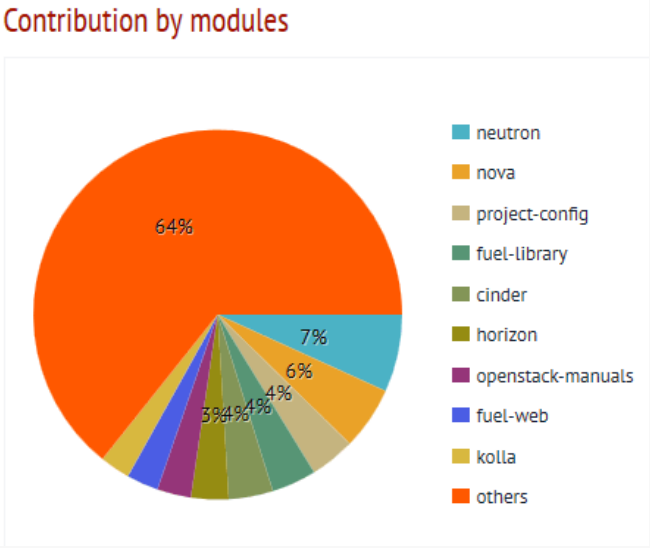
Interactions among components



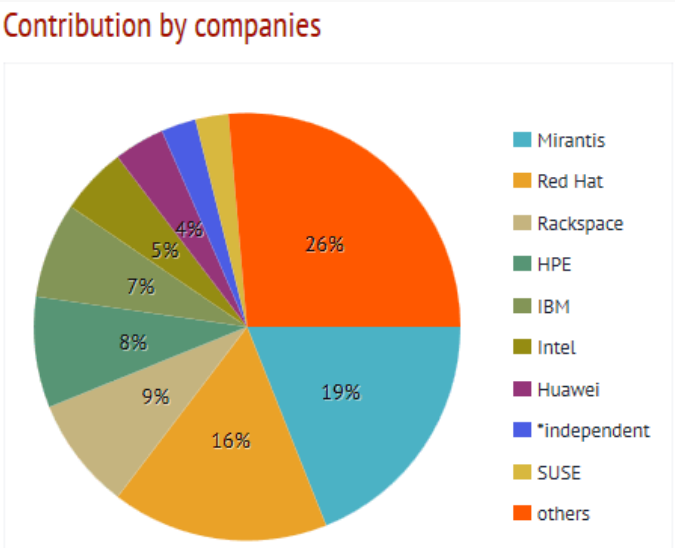
Distribution of projects

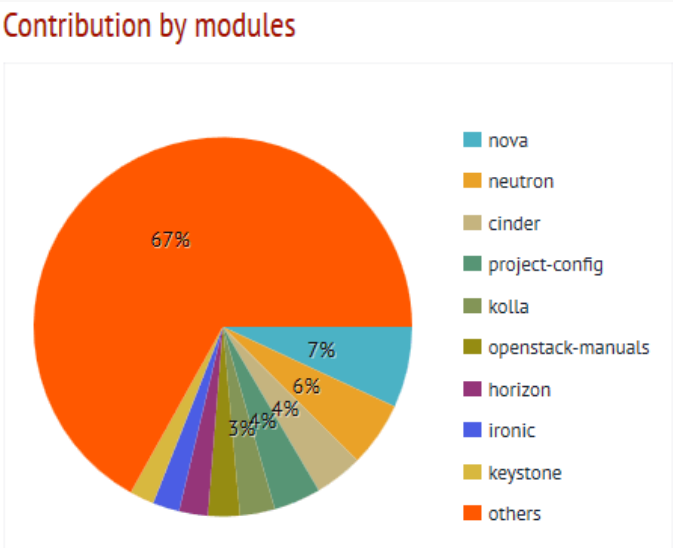
- Mitaka





- Newton

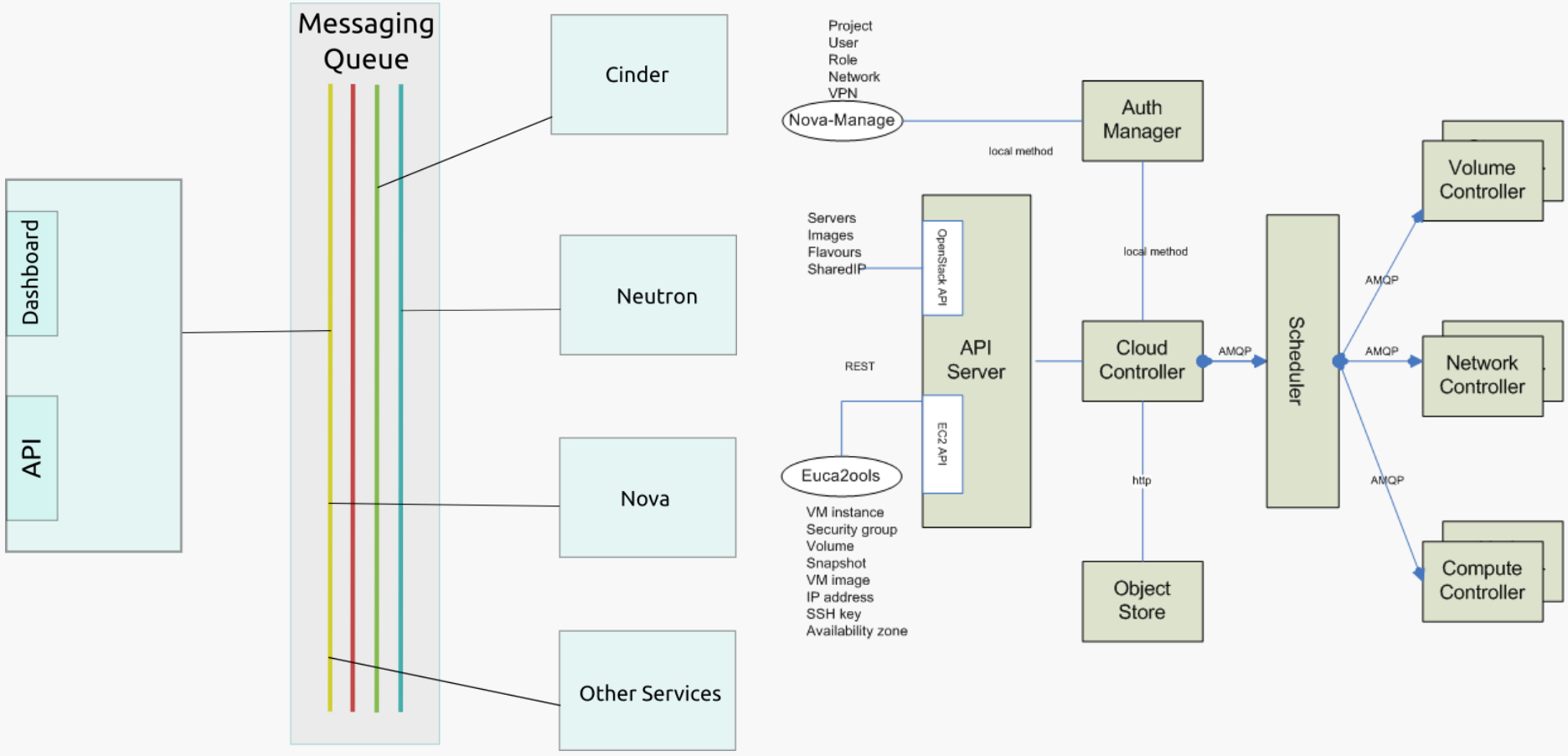




General modules

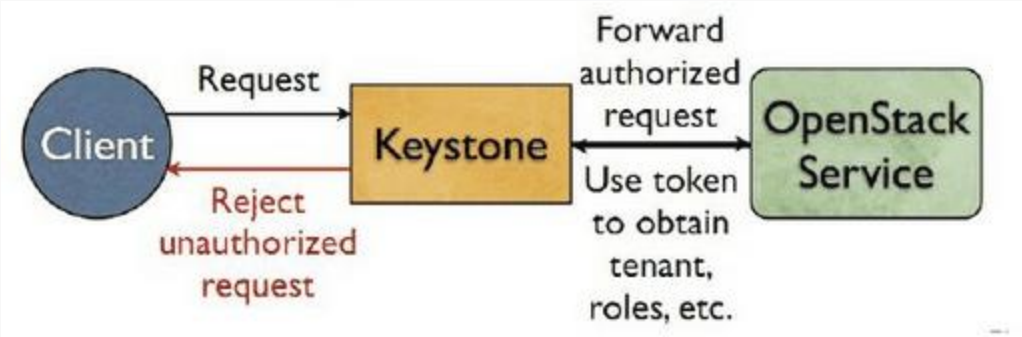
- Message Queue
  - interaction and information exchange between services
- Storage for metadata, configuration data, etc.
  - databases
- Scheduler
  - serving a new VM request

Message Queue

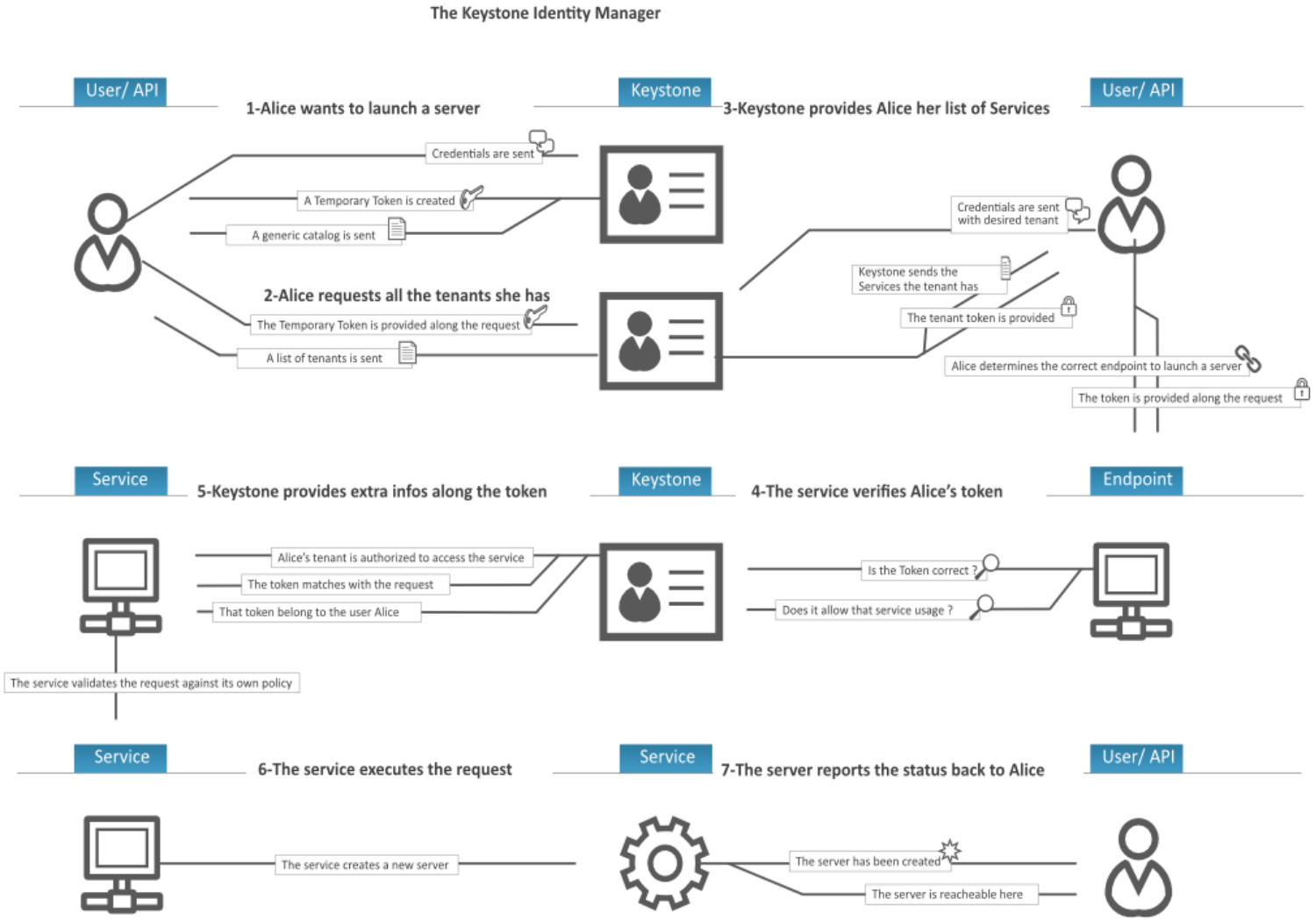


Identity Service: Keystone

- Main Services
  - Identity: auth credential validation and data about *users* and *groups*
  - Resource: provides data about *projects* and *domains*
  - Token: validates and manages tokens used for authenticating requests once a user's credentials have already been verified
  - Service catalog: list of registered services
  - Policy: rule-based authorization engine



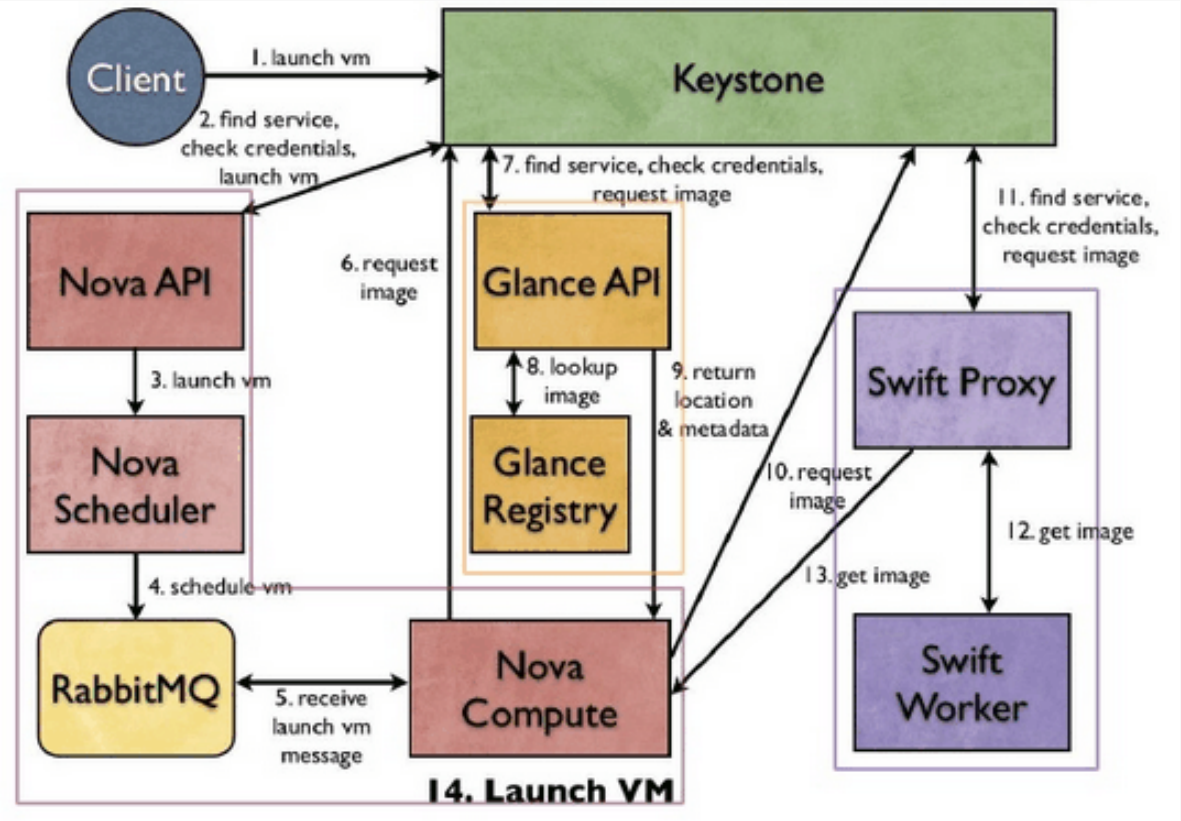
Service initiation and authentication



Compute (Nova)

- Processes
  - nova-api: public interface
  - nova-compute: executing VM instances and managing their lifecycle
    - wide hypervisor support
      - KVM, Xen, XenServer, Hyper-V, etc.
  - nova-volume: managing permanent storage
  - nova-network: networking for VMs
  - nova-schedule: schedule the VM to compute nodes
- Horizontal scaling
  - commodity hardware without special requirements

Starting a VM



Swift (Object storage)

- similar to Amazon S3 (Simple Storage Service)
- scalable, redundant, highly available
- ideal for storing unstructured data that can grow without bound
- replication on multiple hard drives
- storage software for
  - any binary object (data)
    - e.g. VM image, backup, files, etc.
  - can have user *metadata* associated with them
- an object is handled as a unit
  - ideal for data that is mostly read

Cinder (Block level storage)

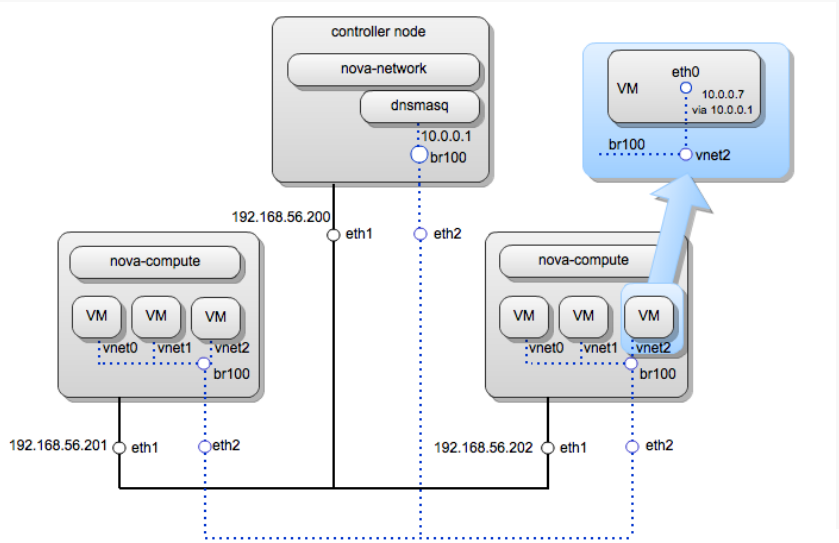
- persistent storage
- typically for file systems: partition, volume
- accesible via API
  - create, delete, attach
  - resize, snapshot
- multiple backend implementation: local server, Ceph, GlusterFS, external storage systems from third-party vendors, etc.
- simpler than Swift, but replication is hard to achieve with multiple vendor backend
- ideal for
  - VM file system
  - database with frequent write

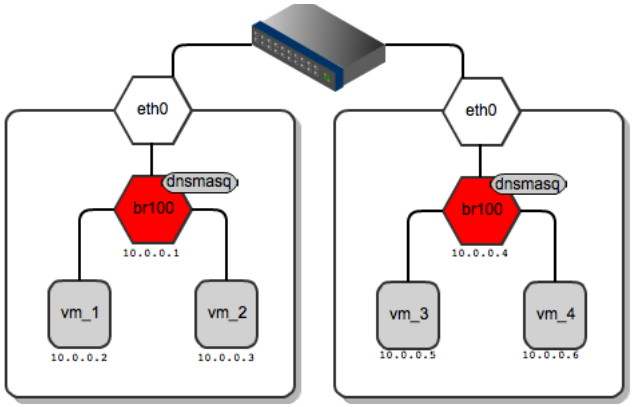
Glance (Image service)

- storage, catalogue and retrieval for disk and container images
  - VM/container templates and associated metadata
- formats: raw, QCOW, VMDK, VHD, ISO, OVF, etc.
- backend service
  - file system
  - Swift
  - Amazon S3

Network architecture

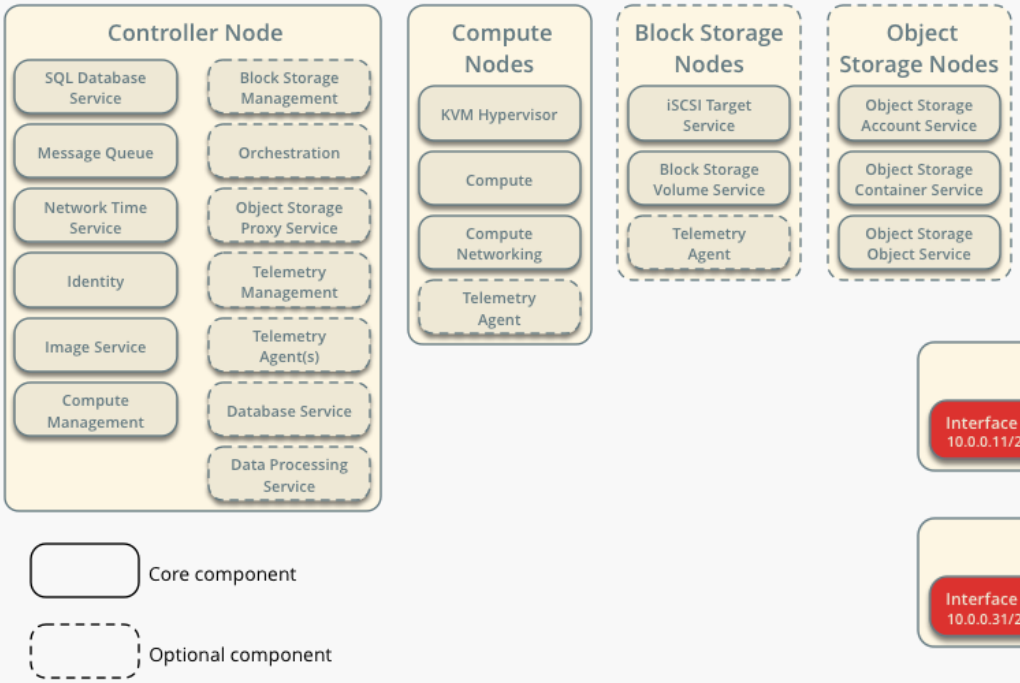
- „Networking in OpenStack is a complex, multifaceted challenge.” /OpenStack Operations Guide/
- Network as a Service
- functions
  - IP addressing
    - static, DHCP
    - floating IP
  - virtual networks
    - flat, VLAN
  - self-service
- alternatives
  - Nova networking / Neutron
  - single-host / multi-host
- Neutron
  - plug-in architecture
  - SDN/OpenFlow



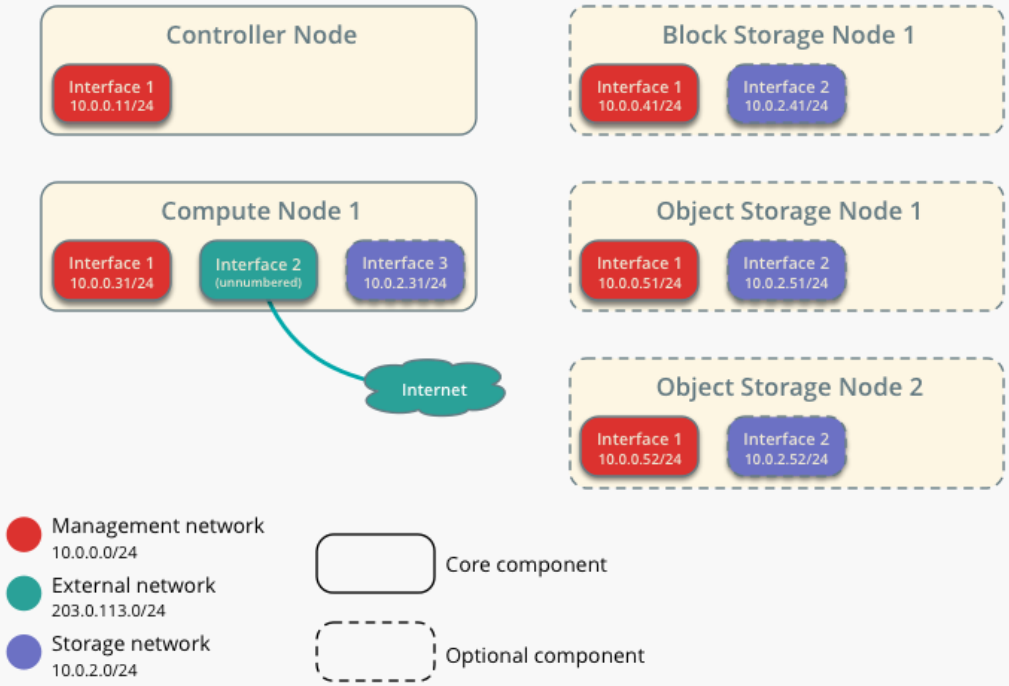


Nova network

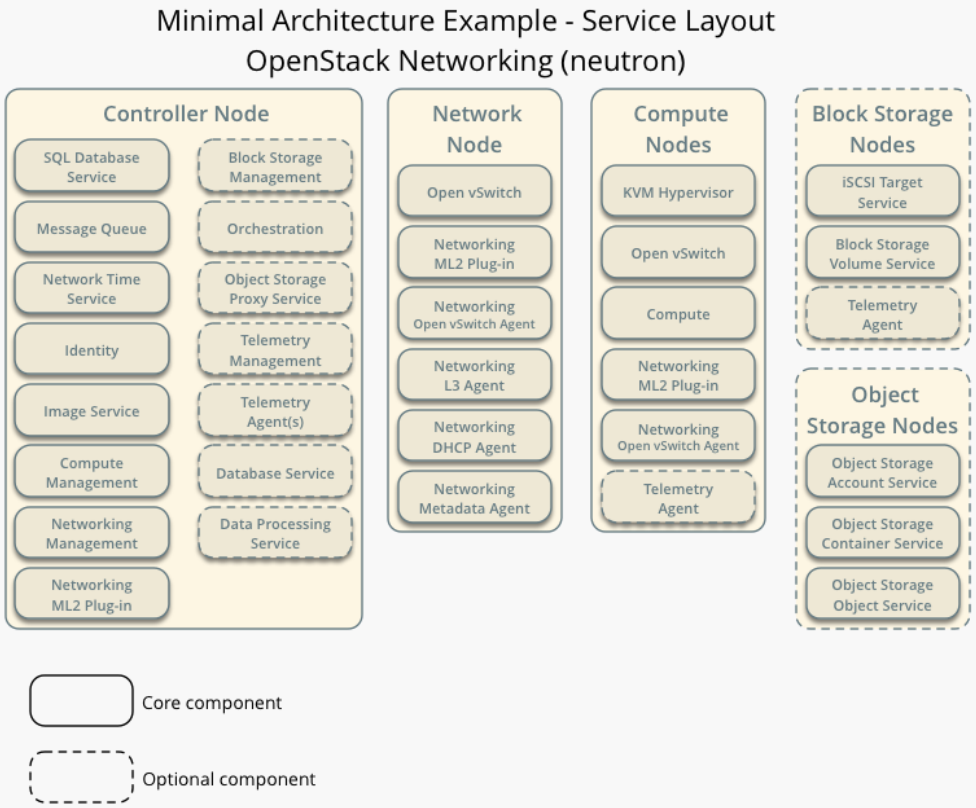
Minimal Architecture Example - Service Layout
Legacy Networking (nova-network)

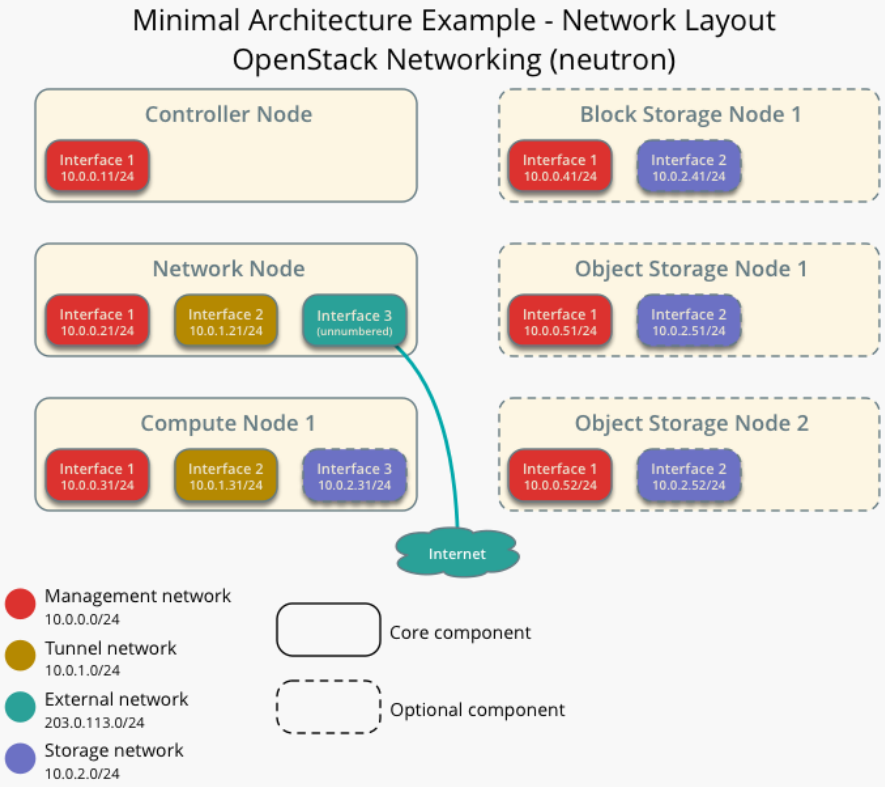


Minimal Architecture Example - Network Layout
Legacy Networking (nova-network)



Neutron network





Networks terminology

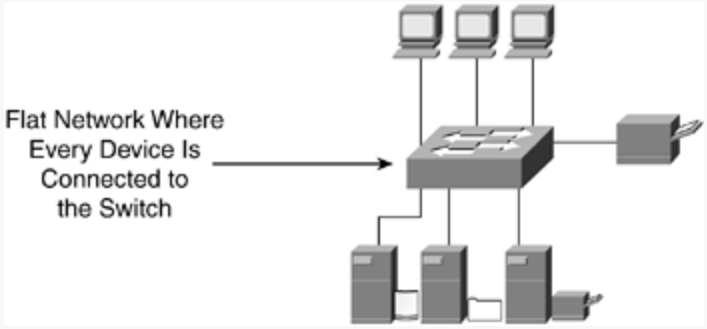
- Internal or management network
  - connects physical nodes
  - for communication between internal components of OpenStack
- External or public network
  - controller external IP address
  - public IP addresses for VMs (floating IP)
    - assigned dynamically to instances

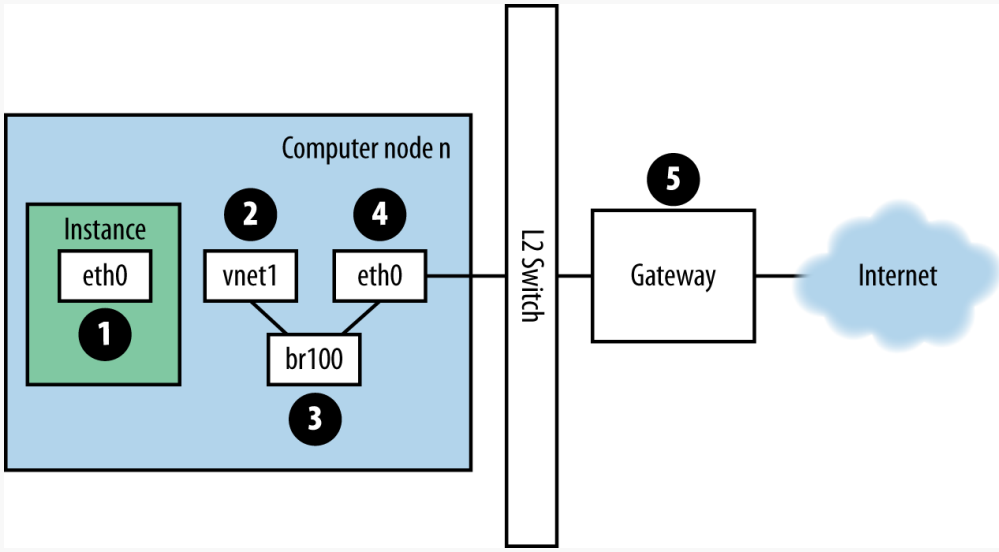
Nova and Neutron Network

- Nova
  - basic networking functions
    - network address translation (NAT), DHCP, DNS
  - only support L2 bridge networking
    - allows virtual interfaces to connect to the outside network through the physical interface
  - limited scalability
    - VLAN, DNS&DHCP (dnsmasq)
- Neutron
  - L3 network, self-service
  - Load Balancing, Virtual IP
  - overlay VLAN tunneling
  - Distributed Virtual Router (from Juno)

Flat Nova Network

- Flat network manager
  - virtual bridge connected to physical node NIC
  - no tenant isolation
  - https://wiki.openstack.org/wiki/UnderstandingFlatNetworking





Network models in Nova networking

Model	Strengths	Weaknesses
Flat	Simple topology: one subnet No DHCP traffic overhead	IP addresses must be configured (file injection at boot time)
FlatDHCP	Relatively easy to deploy Standard networking	DHCP broadcast domain
VLANManager	Each tenant is isolated by its own VLAN	More complex to set up VLAN tagging capable hardware switch DHCP broadcast domains by VLANs Many VLANs to be trunked onto a single port
FlatDHCP mutihost with High Availability	Network failures and DHCP traffic can be isolated to single node Traffic is distributed among compute nodes	More complex to set up Compute nodes typically need IP addresses accessible by external networks Options must be carefully configured for live migration

OpenStack installation alternatives

- OpenStack install guide
  - step-by-step: installing and configuring Linux packages
- General automation tools
  - Chef, Puppet, Juju, Ansible
  - Foreman/QuickStack
- OpenStack specific automation tools / scripted
  - deploying, testing and maintaining
  - hardware discovery
  - provisioning server with GUI
    - Fuel (Mirantis)
    - Ubuntu
      - MaaS + Juju
      - Autopilot: min. 5 server with 2 disks
  - console: RedHat packstack – ssh
- TripleO - OpenStack on OpenStack
- Developer / tester version
  - DevStack
    - setting up a configuration file

OpenStack vs. DevStack

- OpenStack
  - components run as Linux deamons
- DevStack
  - for development and testing
  - minimal configuration
  - startup/shutdown by script
- Deployment options
  - all-in-one physical server / VM
  - multi-node physical servers / VMs

Sketch of 1. Practice

- DevStack Multi-Node Lab
  - http://docs.openstack.org/developer/devstack/guides/multinode-lab.html
- 2 VirtualBox virtual servers: DevStack nodes
  - controller + compute
  - compute
- Netowork model
  - FlatDHCPManager
- Practice
  - According to the Tutorial adapted to the lab environment https://www.mirantis.com/blog/openstack-networking-single-host-flatdhcpmanager/
    - starting VMs on DevStack nodes
    - investigate network architecture

Sources

- http://www.openstack.org
- http://docs.openstack.org
- https://www.mirantis.com/blog/openstack-networking-flatmanager-and-flatdhcpmanager/
- https://www.mirantis.com/blog/openstack-networking-single-host-flatdhcpmanager/